Using Simulated Annealing to Design Interconnection Networks

F. Comellas and **M. A. Fiol**
Departament de Matemàtica Aplicada i Telemàtica
Universitat Politècnica de Catalunya
ETSE Telecomunicació, Girona Salgado s/n
08034 Barcelona, Catalonia, Spain

DMAT Report 05-290, January 1990

Abstract

Simulated annealing is a computational technique reported to give good results in coping with complex combinatorial problems. These problems usually consist of finding a global minimum of a cost function on a (large) set of states (also known as feasible solutions). In this paper we explore modifications to the standard simulated annealing method that we apply to the design of dense interconnection networks, that is networks that have as many nodes as possible for a given maximum number of links from each node to other nodes, and a given maximum distance between all pair of nodes. We are particularly interested in the directed-symmetric case in wich all links have a direction, and the network, roughly speaking, looks the same from any node. In that case each node may execute, without modifications, the same communication software.

The set of states are the different networks and a cost function is defined and is used to accept or reject a new network obtained from a modification of a previously accepted one. The annealing algorithm reduces the 'temperature' parameter and drives the sequence of networks towards an optimum. The results show that with less computational effort than in more traditional approaches we are able to find the best results known and in some cases improve them.

1 Introduction

In the design of interconnection networks two main objectives are to minimize the number of links connecting to a node and the number of nodes that a message must pass through to reach the destination node. In this work we will consider the case in wich the links have a direction, and the same communication software may be used in any node. This is an area of considerable interest for its applications to the design of large interconnection networks, particularly for the construction of massively parallel computers. Other applications include the design of local area networks, the problem of data alignment and the description of some cryptographic protocols.

Such a network can be modelled by an directed graph (or digraph) the vertices of which being associated to the nodes of the network. This digraph should have, therefore, the following properties:

1) The in-degree and out-degree (number of arcs incident to or from a given vertex) should be relatively small.

2) The diameter of the digraph has to be small in relation to the number of vertices.

3) The digraph should be vertex symmetric.

In the context of Graph Theory the problem is an instance of the (Δ, D) problem and consists in the construction of large vertex symmetric graphs and digraphs of a given maximum degree and diameter.

Much work has been done for the undirected case, see [2] for a survey. Some interesting results correspond to Cayley graphs which, as it is known, are vertex symmetric [6]. Actually, most networks related with parallel systems (hypercubes, grids, butterfly networks ...) may be modelled by Cayley graphs. For directed graphs there are well known results concerning bounds [21, 3] and infinite families [4, 14, 15, 13] but the digraphs are vertex symmetric only in special cases.

The search for large digraphs which have the additional property of being vertex symmetric has been considered more recently. Faber and Moore [12], for example, study families of digraphs on permutations and give a table of the largest known vertex symmetric (Δ, D) digraphs. The interest on vertex symmetric digraphs comes from the fact that in the associated network each node is able to execute the same communication software without modifications and in this way these digraphs may be considered for an easy implementation of parallelism.

An extensive part of the search for large graphs and digraphs has been done by using computer methods, sometimes the computer is used for testing for the desirable properties through an exhaustive search. Where exhaustive search was not possible due to the extent of the state space some authors used local search as in [23, 5] or random algorithms, see [10]. The main limitation of local search is that usually leads to local minima. That is because the algorithm only accepts changes that lower the cost function that measures the acceptability of the state. Random algorithms does not perform well if there are few optimal solutions.

In this paper we use another approach based on the so called simulated annealing (SA) technique. The algorithm was first introduced by Kirkpatrich and others, [16], and is derived from a well known work from Metropolis, [18]. In practical optimization problems sometimes it is acceptable to reach a state near the global optimum, because it is better finding a quick good solution to expend a long time to find the global

optimum itself which is only slightly better. The SA algorithm avoids getting stuck at metastable local optima in the search for the global solution because accepts not only changes that lower the objective function but also changes that raise it. The latter are accepted with a probability given by the Boltzmann factor $e^{-\Delta E/kT}$. The range of the parameter T, that is known as *temperature*, depends on the problem considered; the higher the temperature, the more readily is accepted a change that increases the objective function by ΔE. If the model system is first heated and then the temperature is lowered slowly waiting for equilibrium to be established at each temperature, we are performing a simulated annealing procedure. It is precisely such annealing that has the best chance of bringing a real fluid to a true near ground state rather than *freezing* it to a metastable configuration.

In this paper we use simulated annealing for obtaining new largest vertex symmetric (Δ, D) digraphs and most of the digraphs in the table of largest vertex symmetric (Δ, D) digraphs. Section 1 is devoted to notation. In Section 2 we describe the SA algorithms and their application to the solution of problems in Graph Theory. Section 3 shows how it is possible to apply the SA technique to the problem (Δ, D) for vertex symmetric digraphs. Finally in Section 4 we give the main results obtained in relation to the table of largest known vertex symmetric (Δ, D) digraphs

2 Notation and Known Results

A directed graph or *digraph* for short, $G = (V, A)$, consists of a non empty finite set V of elements called *vertices* and a set A of ordered pairs of elements of V called *arcs*. The number of vertices $N = |G| = |V|$ is the *order* of the digraph. If (x, y) is an arc of A, it is said that x *is adjacent to* y or that y *is adjacent from* x, and it is usually written $x \to y$. The *out-degree* of a vertex $\delta^+(x)$ is the number of vertices adjacent from x, the *in-degree* of a vertex $\delta^-(x)$ is the number of vertices adjacent to x. A digraph is regular of degree Δ or Δ-*regular* if the in-degree and out-degree of all vertices equal Δ. A digraph is *strongly connected* if there is a (directed) path from any vertex to every other. The *distance* between two vertices x and y, $d(x, y)$, is the number of arcs of a shortest path from x to y, and its maximum value over all pair of vertices, $D = \max_{x,y \in V} d(x, y)$, is the *diameter* of the digraph. A digraph G is *vertex symmetric* if its automorphism group acts transitively on its set of vertices. A (Δ, D) *digraph* is a digraph with maximum degree Δ and diameter at most D.

The order of a Δ-regular digraph ($\Delta > 1$) of diameter D is easily seen to be bounded by

$$1 + \Delta + \Delta^2 + \ldots + \Delta^D = \frac{\Delta^{D+1} - 1}{\Delta - 1} = N(\Delta, D)$$

This value is called the *Moore bound*, and it is known that, except for $\Delta = 1$ or $D = 1$, there exists no Δ-regular digraphs with $N(\Delta, D)$ vertices and diameter D [21, 3].

The optimization problem considered in this article consists of finding vertex symmetric (Δ, D) digraphs which, for a given diameter and maximum out-degree, have a number of vertices as close as possible to the Moore bound.

A well known infinite family of large (Δ, D)-digraphs, is constituted by the Kautz digraphs [14, 15]. The *Kautz digraph* $K(\Delta, D)$, $\Delta \geq 2$, have vertices labeled with

words $x_1x_2\ldots x_D$ where x_i belongs to an alphabet of $\Delta+1$ letters and $x_i \neq x_{i+1}$ for $1 \leq i \leq D-1$. A vertex $x_1x_2\ldots x_D$ is adjacent to the Δ vertices $x_2\ldots x_Dx_{D+1}$, where x_{D+1} can be any letter different from x_D. Hence, the digraph $K(\Delta, D)$ is Δ-regular and has $\Delta^D + \Delta^{D+1}$ vertices. For $D = 2$ the Kautz digraphs are vertex symmetric. Figure 1 shows $K(2,2)$. In this figure a line represents two opposite arcs.

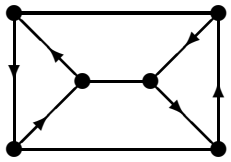

Figure 1: $K(2,2)$, the largest v-s digraph with degree two and diameter two

Faber and Moore in [12] give a family of large vertex symmetric digraphs, which they call $\Gamma_\Delta(D)$. These digraphs may also be defined as digraphs on alphabets in the following way: The vertices are labeled with different words of length D, $x_1x_2\ldots x_D$, such that form a D-permutation of an alphabet of $\Delta+1$ letters. The adjacencies are given by:

$$x_1x_2\ldots x_D \;\rightarrow\; \begin{cases} x_2x_3x_4\ldots x_Dx_{D+1} & x_{D+1} \neq x_1, x_2, \ldots x_D \\ x_2x_3x_4\ldots x_Dx_1 & \\ x_1x_3x_4\ldots x_Dx_2 & \\ x_1x_2x_4\ldots x_Dx_3 & \\ \ldots & \\ x_1x_2x_3\ldots x_Dx_{D-1} & \end{cases}$$

These digraphs have order $\frac{(\Delta+1)!}{(\Delta-D+1)!}$, diameter D and are Δ-regular ($\Delta \geq D$). Note that the digraphs $\Gamma_\Delta(2)$ are in fact the Kautz digraphs $K(\Delta, 2)$.

Most of the entries in the table of large vertex symmetric (Δ, D) digraphs given in [12] correspond to this family.

In this paper we are able of obtaining digraphs with the same order through the SA technique for most of the values of Δ and D, $\Delta < 6$ and $D < 10$

3 Simulated Annealing and Graph Theory

The simulated annealing method comes from the analogy made between the statistical mechanical problem of finding the ground state of a many-body system and finding the minimum or maximum of a given cost function in a combinatorial optimization problem.

If the temperature of the interacting molecules in a liquid configuration is suddenly reduced below its freezing point, the result will be a disordered glassy state with an energy higher than the true crystalline ground state. In fact the molecules are in a local energy minimum.

On the other hand, if the temperature of the liquid is reduced slowly (annealing) waiting for equilibrium to be reached before a new reduction is made, the liquid freezes

to the solid state through a cooling process that leads to the crystalline state, which is the global energy minimum.

In the analogy with the combinatorial optimization problem the parameters being varied are equated with atomic positions in the liquid and its energy is identified with the cost (or objective) function being optimized. The temperature is then defined as a control parameter and is related with the probability that changes which make the state worst will be accepted in order to have a more exhaustive search.

In the algorithm, a change of state that decreases the energy is always accepted; if the energy increases, the change is accepted with a certain probability that depends on the temperature of the system, according to the rule (Metropolis criterion) $e^{-\Delta E/T}$. At a given temperature several exchanges are attempted; then the process is repeated after decreasing the temperature. The system is gradually cooled until it is stopped according to some criteria such as when the number of changes accepted is small and/or the reduction of the energy is not significative. In order to obtain a sufficient statistical set of trials the maximum number of attempts made at a given temperature has to be large enough. The number of successful attempts at each temperature may be also taken in account by the algorithm in order to decide that the state space has been properly searched and the search may finish for this temperature.

Figure 2 correspond to an implementation of the details discused so far.

The most known applications of the SA technique are for scheduling problems like the traveling salesman problem [17] and spatial organization problems like the chip placement [16, 9].

Problems concerning to Graph Theory have been considered more recently, see [1] for a review. For each of these problems there is need of specifying the set of states where to apply the SA method and the cost function.

We give a list of some of these problems where annealing technique has proven useful:

Max (or min) cut problem. Consists of finding, for an edge weighted graph $G(V, E)$, a partition of V, $V = V_0 \cup V_1$ and $V_0 \cap V_1 = \emptyset$, such that the sum of the weights corresponding to the edges joining both sets is maximal (or minimal). The graph partitioning problem correponds to the particular min cut case in which all weights are equal.

Independent set problem. This problem consists of finding an independent set of maximal size $V' \subseteq V$ such that between any two vertices in V' there is no edge.

Graph colouring problem. This is to find a minimal colouring of a graph $G(V, E)$, i.e. a set of l colors and a mapping of V to this set such any two adjacent vertices have different color and the set of colors has minimal cardinality.

Steiner tree problem. Given an undirected connected and weighted graph $G(V, E)$ and a proper subset V' of V, find a minimum-weight tree which spans the vertices of V' and, if necessary, some others.

For these problems although the SA technique is computationally laborious, in fact, it is no more time expensive than most of the best heuristic methods being applied.

Another interesting use of SA in Graph Theory is proposed by Mikhaliov in [19]. From a graph a *melody* is obtained through the adjacency matrix. A SA technique is used to reproduce the graph from its *melody*.

```
INITIALIZE(istart, T0, N0);
k:=0;
i:=istart;
repeat
   for l:=1 to Nk do
      begin
         GENERATE(j from States(i));
         if cost(j) <= cost(i) then i:=j
         else
            if exp((cost(i)-cost(j))/Tk) > random[0,1)
               then i:=j;
      end;
      k:=k+1;
      CALCULATE(Nk);
      CALCULATE(Tk);
until STOPCRITERION.
```

Figure 2: The Simulated Annealing Algorithm.

4 Simulated Annealing and the (Δ, D) Problem

In this Section we show how the SA method may be used in the context of the (Δ, D) problem. In this case the set of states are the different graphs or digraphs and the optimization problem consists of finding graphs or digraphs which, for a given diameter and maximum out-degree, have the largest possible number of vertices. The case that we have considered corresponds to vertex symmetric digraphs. For applying the technique, after deciding on the set of states, there is need for giving a cost function.

We have tested different cost functions. Some of the cost functions are based on previous results using local search [23, 5]. From these results it is known that the diameter is not enough sensitive to changes of the graph and that it is expensive to compute (taking $O(N^2)$). We tested other cost functions used in these works: the number of extremal pairs (two nodes $i, j \in V$ are said to be a *extremal pair* if their distance is equal to the diameter of the graph), the *diameter vector* described in [5] and the mean distance. We found that the easiest to implement, being sensitive enough to changes in the graph, is the mean distance.

Another question refers to the change of states. The more common way, described also in the literature for local search, consists of obtainig a new digraph from an old one by interchanging arcs or *X-crossing*, that is, if in the old digraph we consider two arcs, (i, j) and (k, l), in the changed digraph these arcs no longer exist but there are the

arcs (i, l) and (k, j). It had been proven that by these changes it is possible to reach any state and the state space may be fully explored.

In our study we perform the change of state by changing generators in a Cayley digraph as we describe.

The digraphs we have considered are digraphs on permutations. The vertices of the digraph are different permutations and given a set of generators we contruct the adjacencies through permutation products. More specifically, if $\gamma_1, \gamma_2, \ldots, \gamma_\Delta$ is the set of generators therefore a vertex α is adjacent to vertices $\gamma_1 \circ \alpha$, $\gamma_2 \circ \alpha$, ..., $\gamma_\Delta \circ \alpha$. So, the number of generators gives the degree of the digraph. The starting point in the state space is a set of generators choosen at random and such that the corresponding digraph have de desired order. At each step a generator is replaced at random and the order and cost of the new digraph calculated. If the order is the right one we accept the digraph according to the SA algorithm. Each time that there is an improvement in the cost function, the diameter of the corresponding digraph is tested and the best digraph is recorded. So far we have used this approach for permutations of eight elements.

For the practical implementation we used the algorithms for ranking and unranking permutations described by Stanton and White in [22]. These algorithms enables us to store the graphs in a very compact way and to access any permutaion quickly. The resulting program is small and efficient.

5 Conclusion

The following tables sumarize the results obtained. The small size values of Table 1 are the best best vertex symmetric digraphs as reported in [12]. The large size values were obtained by SA. As we see they reproduce or improve (in bold) those obtained by Faber and Moore. More improvements to this table have been found by using another different approach based on some generalizations of results from Conway and Guy, see [7, 8].

Table 2 shows the generators of the symmetric group that were obtained through the algorithm.

D / Δ	2	3	4	5	6	7	8	9
2	6 ·	10 10	20 20	24 24	60 60	120 ·	171 ·	336 ·
3	12 ·	24 ·	60 60	120 120	168 ·	720 720	· ·	2 520 2 520
4	20 ·	60 ·	120 120	360 360	720 720	2 520 2 520	5 040 5 040	· ·
5	30 ·	120 ·	360 ·	840 840	2 520 2 520	5 040 5 040	· **40 320**	· ·

Table 1: Best Vertex Symmetric Digraphs Obtained by Simulated Annealing.

```
TAULA DE RECORDS:

Grafs sobre permutacions.
Millors valors trobats per
ANNEALING o busqueda exhaustiva
====================================
====================================
GRAU 2 / DIAM 4 ==> ORDER 20
 20 >>    4   6  -->     4
4 :: 15234 ....>(1)(2543)
6 :: 51243 ....>(1532)(4)
++++++++++++++++++++++++++++
GRAU 2 / DIAM 5 ==> ORDER 24
 24 >>    4  12  -->     5
4 :: 15234 ....>(1)(2543)
12 :: 14253 ....>(1)(2453)
++++++++++++++++++++++++++++
GRAU 2 / DIAM 6 ==> ORDER 60
 60 >>    5  23  -->     6
5 :: 51234 ....>(15432)
23 :: 41532 ....>(14352)
++++++++++++++++++++++++++++
GRAU 2  / DIAM 11 ==> ORDRE 720
0.00100 11  5842
++++++++++++++++++++++++++++++
4690 :: 2146537 ....>(12)(346)(5)(7)
4566 :: 2641573 ....>(126734)(5)
++++++++++++++++++++++++++++++
======================================
======================================
GRAU 3  / DIAM 4 ==> ORDRE 60
 60 >>    5  15  63  -->     4
  5 :: 51234 ....>(15432)
 15 :: 51423 ....>(15342)
 63 :: 43521 ....>(14235)
++++++++++++++++++++++++++++++
GRAU 3  / DIAM 9 ==> ORDRE 2520
2520->  9
2520 >> 3583  35   7  -->     9
3583 :: 5623471 ....>(1543267)
35 :: 7162345 ....>(1753642)
7 :: 7123456 ....>(1765432)
++++++++++++++++++++++++++++++
GRAU 3  / DIAM 10 ==> ORDRE 5040
5040 >>    2 31420227  -->    10
2 :: 1234576 ....>(1)(2)(3)(4)(5)(67)
314 :: 1726543 ....>(1)(273)(46)(5)
20227 :: 7246351 ....>(17)(2)(3465)
====================================
====================================
GRAU 4  / DIAM 4 ==> ORDRE 120
120 >>    3   7  41  50  -->     4
  3 :: 12534 ....>(1)(2)(354)
  7 :: 15243 ....>(1)(253)(4)
 41 :: 31245 ....>(132)(4)(5)
 50 :: 31425 ....>(1342)(5)
++++++++++++++++++++++++++++++++++
GRAU 4  / DIAM 5 ==> ORDRE 360
0.00100  5  1427
++++++++++++++++++++++++++++++
701 :: 265134 ....>(1264)(35)
339 :: 453612 ....>(14625)(3)
317 :: 364512 ....>(1345)(26)
217 :: 153246 ....>(1)(254)(3)(6)
++++++++++++++++++++++++++++++
```

```
====================================
====================================
GRAU 4 / DIAM 6 ==> ORDRE 720
0.00100  6  3290
++++++++++++++++++++++++++++++++
244 :: 316245 ....>(136542)
95 :: 541263 ....>(1563)(24)
537 :: 236415 ....>(12365)(4)
694 :: 521634 ....>(153)(2)(46)
++++++++++++++++++++++++++++++++
=====================================
=====================================
GRAU 5 / DIAM 6 ==> ORDRE 2520
0.00100  6 12177
++++++++++++++++++++++++++++++++
363 :: 1264573 ....>(1)(2)(367)(4)(5)
2057 :: 6314572 ....>(16723)(4)(5)
3197 :: 3572146 ....>(1376425)
2137 :: 6734125 ....>(16275)(3)(4)
369 :: 1672453 ....>(1)(2654)(37)
++++++++++++++++++++++++++++++++
=======================================
=======================================
GRAU 5 / DIAM 8 ==> ORDRE 40320
0.00100  8265290
++++++++++++++++++++++++++++++++
Order: 40320  Degree: 5 Diameter:   8
      1810 :: 51627483 ....>(15783642)
     31637 :: 75284316 ....>(17)(254863)
      6720 :: 41235678 ....>(1432)(5)(6)(7)(8)
      8952 :: 81564372 ....>(182)(3546)(7)
       614 :: 71823564 ....>(17653842)
++++++++++++++++++++++++++++++++
=============================================
=============================================
```

Table 2: Generators of S_n.

References

[1] E. Aarts and J. Korst. *Simulated Annealing and Boltzmann Machines.* John Wiley & Sons. Chichester, 1989.

[2] J-C Bermond and C. Delorme and J-C Quisquater. Strategies for interconnection networks: Some methods from Graph Theory. *J. Parallel and Distrib. Comp.* **3**, pp. 433-449, 1986.

[3] W.G. Bridges and S. Toueg, On the impossibility of directed Moore graphs. *J. Comb. Theory B.* **29**, pp. 339-341, 1980.

[4] N.G. de Bruijn, A combinatorial problem. *Koninklijke Nederlandse Academie van Watenschappen Proc.*, **A49**, pp. 758-764, 1946.

[5] G.E. Carlsson, J.E. Cruthirds, H.B. Sexton, C.G. Wright and M.J. Shensa. A local search algorithm for constructing regular graphs of small diameter. Submitted IEEE, 1984?.

[6] D.V. Chudnovsky, G.V. Chudnovsky and M.M. Denneau, Regular graphs with small diameter as models for interconnection networks. *Proc. Third Int. Conf. on Supercomputing, ICS'88*; Boston, May 16-20 1988, pp. 232-239.

[7] F. Comellas and M.A. Fiol. Vertex symmetric digraphs with small diameter. *Discrete Mathematics*, submitted, 1990.

[8] J.H. Conway and M.J.T. Guy, Mesage graphs. *Ann. of Discrete Math.*, **13**, pp. 61-64, 1982.

[9] F. Darema, S. Kirkpatrick and V.A. Norton. Parallel algorithm for chip placement by simulated annealing. *IBM J. Res. Develop.*, **31**, pp. 391-402, 1987.

[10] K.W. Doty. New designs for dense processor interconnection networks. *IEEE Trans. Comp.* **C-33**, pp. 447-450, 1984.

[11] B. Elpas, Topological constrains on interconnection limited logic. *Proceedings 5th Symposium on Switching Circuit Theory and Logical Design*, **5-164**, pp. 133-197, IEEE, New York, 1964.

[12] V. Faber and J. W. Moore, *High-degree low-diameter interconnection networks with vertex symmetry: the directed case.* Technical Report LA-UR-88-1051, Los Alamos National Laboratory, Los Alamos, New Mexico, 1988.

[13] M.A. Fiol, J. L. A. Yebra and I. Alegre, Line digraph iterations and the (d, k) digraph problem. *IEEE Trans. Comp.* **C-33**, pp. 400-403, 1984.

[14] W.H. Kautz, Bounds on directed (d, k) graphs. *Theory of cellular logic networks and machines*, AFCRL-68-0668, SRI Project 7258, Final Report, pp. 20-28, 1968.

[15] W.H. Kautz, Design of optimal interconnection networks for multiprocessors. *Architecture and Design of Digital Computers*, Nato Advanced Summer Institute 249-272, 1969.

[16] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi, Optimization by Simulated Annealing. *Science*, **220**, 671-680, 1983.

[17] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys. *The Traveling Salesman Problem. A Guided Tour of Combinatorial Optimization.* John Wiley & Sons. Chichester, 1985.

[18] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller, *J. Chem. Phys.*, **21**, 1087, 1953.

[19] A.S. Mikhailov Graph identification by simulated annealing *J. Phys. A: Math. Gen.* **21**, pp. L487-L490, 1988.

[20] R.H.J.M. Otten and L.P.P.P. van Ginneken. *The Annealing Algorithm.* Kluwer Academic Publishers, Boston, 1989.

[21] J. Plesník and Š. Znám, Strongly geodetic directed graphs. *Acta Fac. Rerum Natur. Univ. Comenian., Math. Publ.* **29**, pp. 29-34, 1974.

[22] D. Stanton and D. White. *Constructive Combinatorics.* Springer-Verlag, New York, 1986.

[23] S. Toueg and K. Steiglitz, The design of small-diameter networks by local search. *IEEE Trans. Comp.* **C-28**, pp. 537-542, 1979.